LAWRENCE
LIVERMORE
NATIONAL
LABORATORY

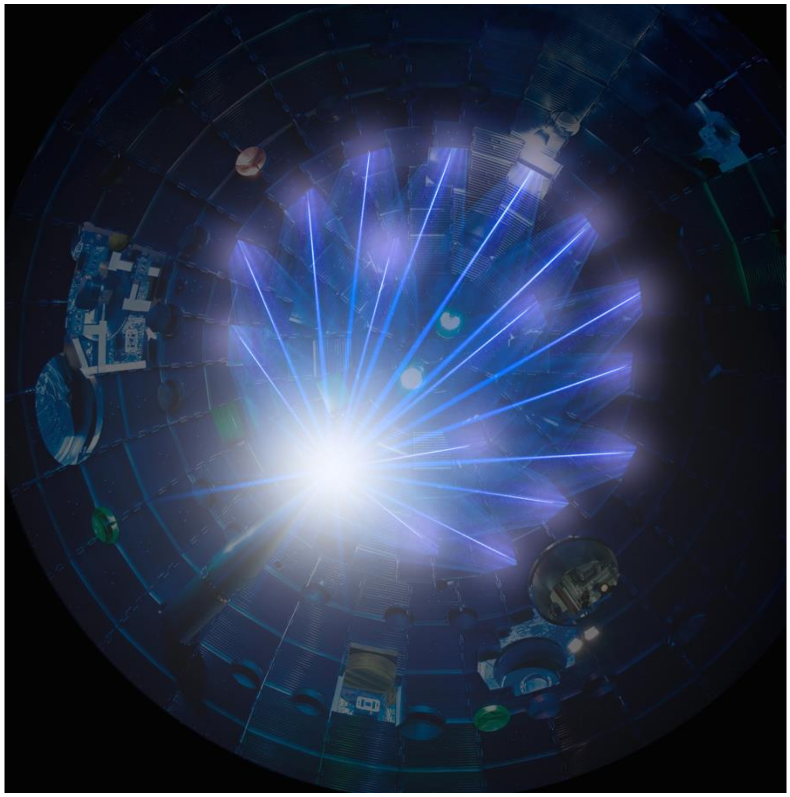
LLNL-TR-667131

NIF Experiment Editor Redesign

D. Speck

February 10, 2015

This work performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344.



NIF Experiment Editor Redesign

Setting up for the future

Douglas Speck

SYS 625 Fundamentals of System Engineering
Dinesh Verma
January 30, 2015

Table of Contents

NIF Experiment Editor Redesign 1
Setting up for the future 1
Executive Summary 1
Mission Description 2
System Operational Context and Reference Operational Architecture 10
System Drivers and Constraints 15
Operational Scenarios 16
Implementation Concepts and Selected and Rationale 22
Proposed System Operational Architecture 22
System Requirements 22
Organizational and Business Impact 22
Risks and Technology Readiness Assessment 22

Douglas Speck

SYS 625 Fundamentals Of System Engineering
Dinesh Verma
January 30, 2015

NIF Experiment Editor Redesign

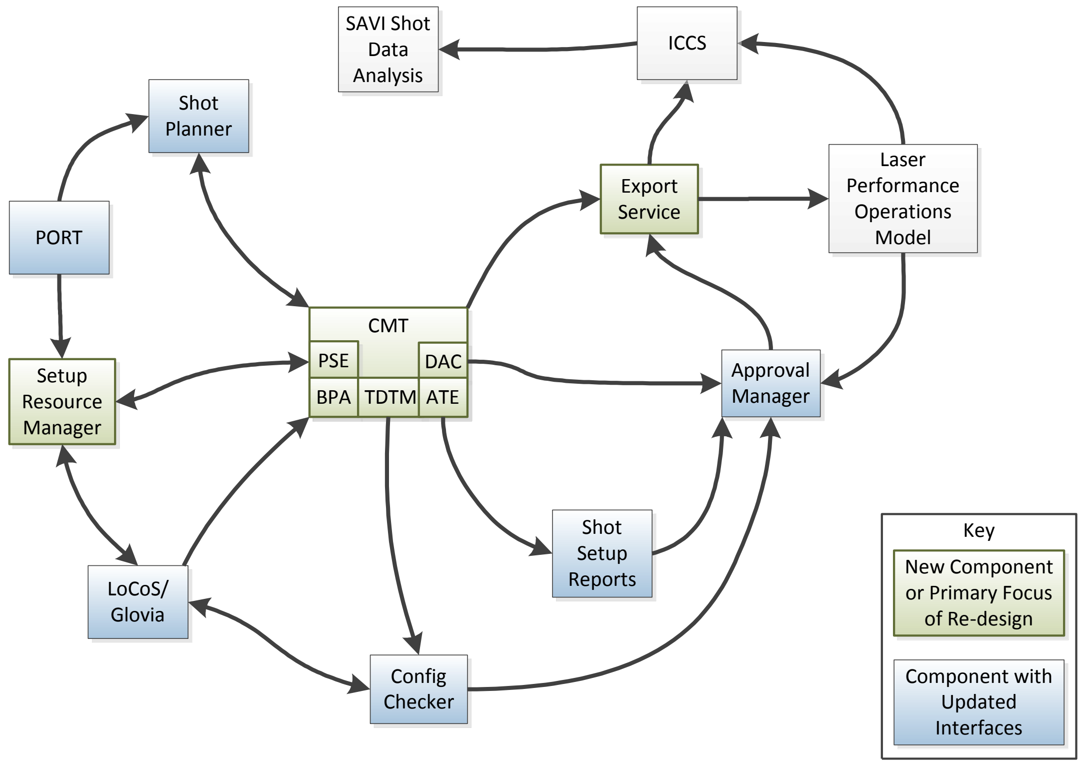
Setting up for the future

Executive Summary

The Campaign Management project at the National Ignition Facility is embarking on a major redesign of the software used by experimentalists to setup experiments on the NIF. The current editor was architected over a decade ago and optimized for a different purpose than supporting production science campaigns. It has a steep learning curve and lacks many desirable features. Furthermore, its underlying technology stack and architecture do not support the kind of routine capability evolution that represents current and future growth needs of the Program.

The principal stakeholders are the physicists who are users of the editing application, but the entire integrated suite of Campaign Management software has a broad customer base spanning experimentalists, NIF sub-system subject matter experts, planners, and shot operations staff.

A recent DOE-mandated review of NIF activity, conducted with the goal of improving the shot rate at NIF, identified the experiment editor as a clear barrier to a more productive and efficient experiment lifecycle, exposing unnecessary levels of detail, lacking adequate automation, and lacking adequate integration with related activities and data sources in what should be a shared data environment.



Planned Software System Architecture

Mission Description

The National Ignition Facility (NIF) experiment lifecycle can be decomposed into a handful of phases. For example: Proposal, Review, Planning, Setup, Approval, Readiness, Shot, and Analysis (other decompositions are possible). Given the relative youth of the NIF (science campaigns began in 2010), these activities are at varying levels of maturity and are under constant pressure to evolve in order that NIF can produce the biggest scientific bang for the buck.

In the first few of those lifecycle phases, an experiment appears in relatively abstract form defined by goals, theoretical analysis results and primary experimental characteristics. In the Setup phase it is transformed by a small team of physicists from that abstract representation into an explicit, detailed, low level representation containing most of the inputs needed by the Integrated Computer Control System, ICCS, to execute the laser shot cycle and conduct the actual experiment.

Setup is performed using a custom editing application called the Campaign Management Toolset (CMT). CMT is a Java application that consists of a desktop client, the "CMT" users identify with, and a backend server that primarily manages two interfaces: the interface between all desktop clients and the underlying database where CMT data is stored, and the interface to ICCS through which experiments are exported to the control system. In fact CMT is the core application in a Campaign Management suite that includes a variety of specialized helper applications and associated workflow managers that operate in an integrated manner around the primary CMT product: an XML document containing the thousands of parameters and settings needed by ICCS for the performance of a single experiment.

CMT was architected approximately 12 years ago with the primary purpose of enabling the NIF laser to be commissioned. As such it exposes a low level view of the experiment setup, maximizing the user's ability to "turn every knob" at the cost of a dizzying level of complexity and corresponding difficulty in completing the setup. One of the principal themes to come out of the DOE-mandated 120-day review of shot operations completed last year, conducted with an eye to improving efficiencies and shot rate at the Facility, was the barrier CMT's complexity represents to realizing a reduction in the time required to setup an experiment.

As the experimental program at the NIF matures, new priorities emerge for managing the data associated with an experiment. Improving the shot rate, one of the holy grails of NIF operations, demands improved planning and an integrated sequencing of experiments designed to minimize shot-to-shot transition costs, measured in time and labor. Many aspects of the experiment definition are now seen to properly belong not to the experimenter but to planning and operations activities. For example, an experimenter might ideally enjoy complete freedom to specify which target chamber diagnostic devices are employed to gather data during her shot. However, given the high cost of exchanging primary diagnostics, the operations staff sees great benefit in minimizing such transactions. Therefore the planning activity attempts to optimize shot-to-shot operations by owning experiment sequencing and final control of the particular diagnostic hardware that is employed on experiments. Unfortunately, CMT does not now "share" control of the data it manages, opening the door to changes in setup not reviewed and approved by planning and operations, which can lead to frustration, disarray, and even the occasional scrubbed shot (a potentially seven-figure mistake).

When CMT was designed, managing the one-of-a-kind NIF laser was the critically important task. For all of its size and complexity, the 192-beam laser lends itself to a setup information hierarchy that is highly structured and repetitive, and CMT's architecture reflects that. Today however, the laser evolves very little and the overwhelming driver for change in experiment setup is the non-stop evolution of NIF targets and target chamber diagnostic devices. Existing target diagnostics (TDs) are routinely improved and new ones are introduced to the NIF several times a year, and each TD requires a unique setup interface in CMT. The current architecture tightly binds GUI display elements with setup data elements from the experiment XML document. This is a powerful abstraction for doing a few basic things consistently across the application, but it becomes a barrier when trying to manage the variety of functional relationships among the data in these unique TD setups. That CMT still functions in daily use a dozen years after its design was laid down, when full experimental operations did not commence for seven years following it's initial deployment, is a testament to the vision of its designers and the extensibility of its design. Now though, after five years of science campaigns, we can see clearly where the current architecture and embedded technology stack is not optimal for the current development demands and can be a barrier to timely response to typical feature requests from the NIF program participants (let alone the atypical ones).

Thus for powerful reasons arising from both the user and development communities, we have undertaken development of a next-generation experiment setup editor. Our mission is to:

> Realize a dramatic and ongoing improvement in efficiency for setting up experiments on the NIF by

- Automating setup where settings can be derived from higher-level parameters, physics goals, or well-known relationships
- Facilitating setup through comparative views of related setup data (internal) and related contextual data (external)
- Simplifying setup and setup workflow using appropriate levels of data fluidity and locking and removing data sub-domains that belong to other NIF activities
- Fielding an underlying architecture that enables faster and more robust evolution

The key stakeholders for this mission are:

1. Experimenters – active stakeholders

a. Experimentalist roles

   i. Shot Responsible Individual (Shot RI) – The experimental physicist who "owns" an experiment and is responsible for driving it through its lifecycle. Uses CMT to setup the laser and target systems; may set up one or more target diagnostics.

   ii. Target Diagnostic Responsible Scientist (RS) – The subject matter expert for a particular TD; typically performs the TD setup and/or review & approval of a setup for the relevant diagnostic.

   iii. Laser subject matter experts (Beamline Integrated Performance, "BLIP" team) – responsible for managing and monitoring laser performance; use CMT to setup non-target experiments for which the goal is to provide data about the current state of the laser. In addition to executing there own laser-performance related shots, BLIP are also customers of laser setup data for all target shots, providing approval of each experiment's laser configuration.

   iv. Target and Laser Interaction (TaLIS) subject matter experts – responsible for issues related to machine safety and damage estimation deriving from all shot-related events and activities inside the NIF target chamber. This includes setup and alignment as target and diagnostic positioners traverse space inside the chamber, reflected laser light during a shot, and debris and shrapnel from an implosion. TaLIS run their own experiments to analyze alignment and energy-related issues, and as approvers for relevant sections of the setup for all target shots are also customers of setup data for those experiments.

b. Expectations

   i. Make it easier and faster to setup an experiment by:

      1. [Sacred] Removing items from Shot RI responsibility that are not central to their goals (e.g., setup of beams for a target shot that are not going all the way to the target chamber and thus will not affect the experimental outcome; setup items that should only be accessed by specialists such as laser subject matter experts).

2. [Sacred] Automating setup where parameter values can be reliably derived from other parts of the setup or known high level goals of the experiment

3. Assist in dealing with the complexity of large selection pools through appropriate classification, filtering, and smarter views

4. [Sacred] Assist in setup decision-making by providing easy access to relevant data sources and intrinsically useful views (e.g. a timing plot to directly expose the temporal relationships between beam pulses, detector sweeps, timing pulses, etc; direct views of planning data, proposal data).

5. [Sacred] Improve flexibility and ease of applying existing setups, be they defined templates or sub-selections from other experiments, into experiments under development

6. Lower the cost of making late changes to an experiment when those changes don't carry risks to realizing a successful experimental outcome or risks to machine safety

7. [Sacred] Enable access to "as shot" experiment configurations within CMT to reflect changes made during the shot cycle. Information flow is currently one way and there's no way to directly pull changes back into CMT.

2. Operations – active stakeholders

   a. Roles

      i. Shot operations Directors, Coordinators, and Operators – Prepare the facility for each shot and execute the shot cycle. These are usually not users of CMT itself but often access other Campaign Management applications and are routine customers of many setup data reports and workflow status reports.

      ii. Target and Cryogenic staff – manage planning, production, and installation of targets. Typically do not use CMT but are both providers and consumers of experiment setup data and use Campaign

Management applications other than CMT; manage data resources that CMT interfaces with.

b. Expectations

i. Minimize approval workflow interruptions

ii. Update target system dependencies and logic to reflect current workflow and constraints. System configurations associating targets, shields, shrouds, gases, and pressures are too restrictive

3. Planning – active stakeholders

a. Roles

i. Manage the long- and short-term sequencing of experiments and related resource acquisition and allocation; a data provider for CMT and owner of data resources CMT interfaces with.

b. Expectations

i. [Sacred] Support planning activity ownership of relevant setup parameters with suitable bi-directional visibility and change management

4. Program Management – passive stakeholders

a. Roles

i. NIF User Office - Own responsibility for making the NIF available and usable for all prospective experimenters; have a vested interest in all user tools of which CMT is a critical component, so own user-oriented requirements for Campaign Management applications.

ii. Stockpile Stewardship, Inertial Confinement Fusion (ICF), High Energy Density (HED) Science research program managers – Line activities providing resources and sponsored experimenters (i.e. Shot RIs) to conduct directed research. Interact closely with NIF management and NIF User Office to represent their programs' experimental needs, especially as identified by their Shot RIs.

b. Expectations

    i. [Sacred] Unified (virtually or in actuality) experiment planning & setup databases with appropriately segregated data ownership

    ii. [Sacred] Elimination of barriers to efficient experiment lifecycle including improved setup, enter-once/share-everywhere experiment data management; i.e., happier, less frustrated Shot RIs. These are part of a broader thrust for improvements across the pre-shot lifecycle that encompasses proposal, planning, setup, and approval. Only these last two are the responsibilities of Campaign Management.

    iii. [Sacred*[*]] Two-hour setup time for experiments being developed "from scratch", i.e., not via cloning from an existing experiment

5. Engineering – active stakeholders

    a. Roles

        i. TD Engineering teams responsible for initial development and evolutionary updates of target diagnostic systems. Engineering owns the requirements for specification of TD setup and own the sets of setup options that CMT must expose to experimenters.

        ii. Laser Engineering responsible for laser-related hardware. Ultimate owner for requirements related to the laser itself, but these usually manifest for Campaign Management as requirements from the ICCS development team when, for example, laser changes result in new setup options.

    b. Expectations

        i. Improved turnaround for revisions to existing TDs

        ii. Improved management of setup option selection lists: suitable classification, filtering, and display of choices; quicker update cycles for changing options,

---

[*] It's not clear if this is sacred, or if it should be. That is, there doesn't seem to be any analysis supporting this as a reasonable goal. But someone said it, and now it's in the dialog.

6. Software Activities – active stakeholders

    a. Software projects

        i. ICCS – the laser control system interfaces with CMT via a relational database to import the data needed for each experiment.

        ii. Laser Performance Operations Model (LPOM) – critical modeling tool used to understand how the laser will perform for a particular shot given the current state of NIF laser optics and the requested performance of each beam defined in the CMT setup – a CMT data consumer. LPOM receives data and execution requests from CMT via both an XML file-based setup definition for pre-export analysis and from a relational database-based setup definition for post-export final analysis prior to shot execution.

        iii. Post-shot Analysis - typically receive needed data from the shot archive but occasionally require access to CMT setup data

        iv. Shot Planner – software tool for the Planning activity to designate planning-owned experiment data; both a data provider and consumer with respect to CMT

        v. LoCoS/Glovia – parts management and facility status tracking for the entire NIF. Both a data provider and consumer with respect to CMT. Glovia (i.e. its underlying database) is the ultimate owner of all part and serial numbers.

        vi. PORT – Database application that is the principal repository of target request, build, and status information; a data provider for Campaign Management. PORT has been incrementally absorbing target-related functionality from a CMT helper application, Target Selection Manager, and will ultimately replace it (to be clear, functionality inherited from TSM does and will represent only a very small part of PORT overall functionality).

    b. Expectations

        i. [Sacred] Web-based UI. Web-based applications have been the execution architecture of choice within Shot Data Systems for several years. Although this could be considered an implementation choice, it is in fact accepted on both the software development side and the end user side as a given (see also Oracle stakeholder below).

   ii. Eliminate "dual-master" issues arising from separate definitions of data within Campaign Management that are ultimately owned elsewhere; in particular, part numbers that are maintained as part of CMT selection options data but are actually defined within Glovia

   iii. [Sacred] Enable change-request workflows between CMT and other data owners for data exposed within CMT but not owned by experimenter roles

   iv. Eliminate TSM by migrating remaining functionality into PORT

7. Oracle – passive stakeholder

   a. Roles

      i. Java language standard owner

      ii. Oracle Relational Database Management System owner

      iii. WebLogic web services container owner

   b. Expectation

      i. [Sacred] Eliminate dependencies on Java WebStart and Java Network Launch Protocol (JNLP)

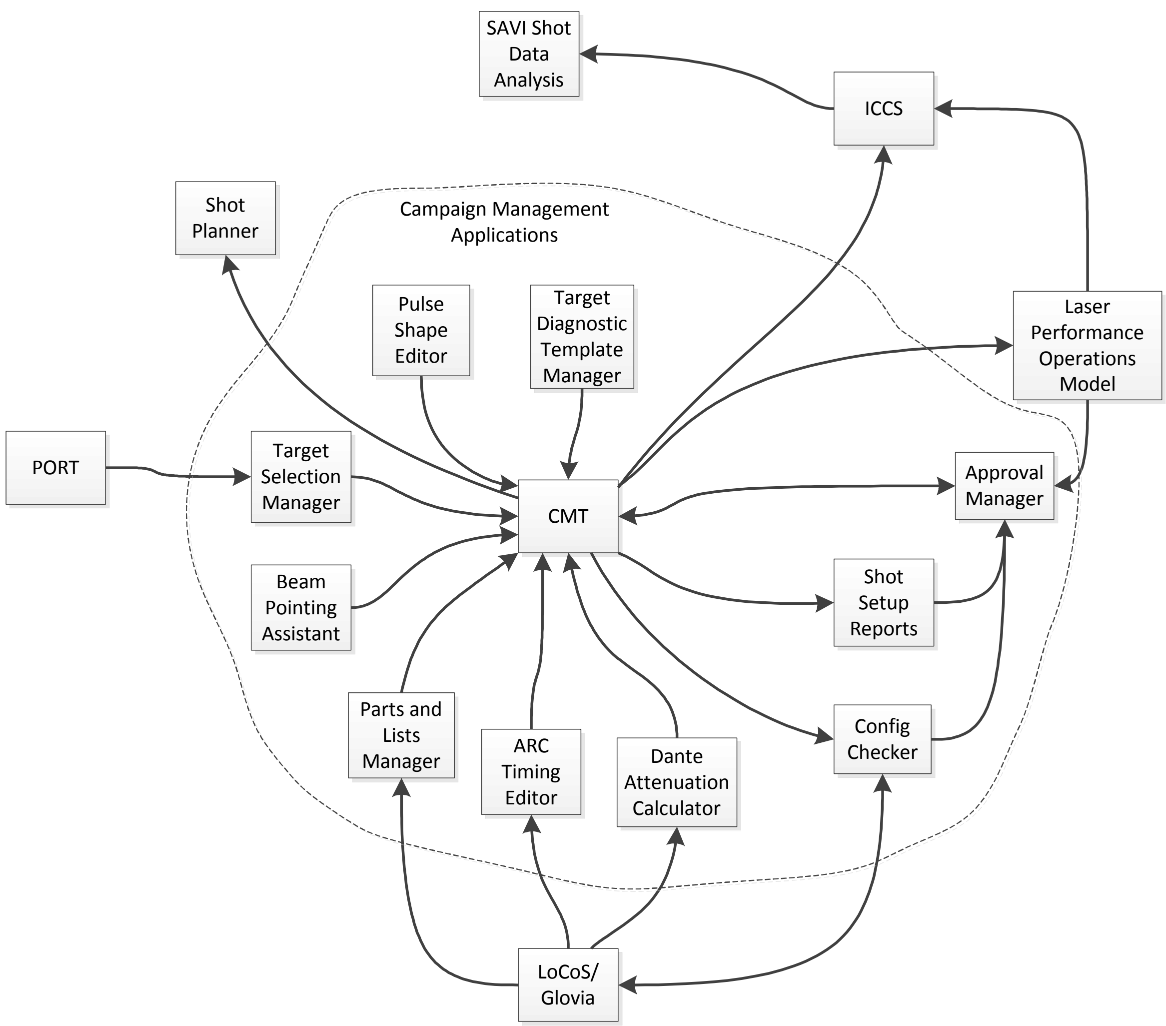
System Operational Context and Reference Operational Architecture

Current System Architecture

The current Campaign Management software suite includes three applications that directly support different aspects of the experiment lifecycle, and another eight assistant applications. The primary application, and central focus of this re-engineering effort, is the experiment editor, CMT. CMT is supported by these assistant applications:

- Parts and Lists Manager (PLM) – a database front end through which we manage the options available in each setup selection in the CMT UI. The contents managed are defined mostly by the engineering teams responsible for the supported systems (e.g., each target diagnostic). Some calibration data maintained in LoCoS and part data maintained in Glovia are exposed to CMT via PLM, so in that sense PLM also provides a level of isolation between two large critical software systems.

- Target Selection Manager (TSM) – performs a roughly analogous function to PLM but for target systems data, including targets, shields, shrouds, and mask kits. Most of that data is now actually maintained in PORT (which is not part of Campaign Management) since TSM and PORT partially merged two years ago, but TSM exposes its views of PORT data to CMT as it always has so that the TSM/CMT interface is unchanged.



Legacy Software Context

- Beam Pointing Assistant (BPA) – BPA is a specialized editor that lets Shot RIs develop and manage beam aimpoint coordinates for the 192 NIF beams.
- Pulse Shape Editor (PSE) – Another specialized editor, PSE lets users define beam pulse shapes as arbitrary spline curves, saving them as named files that are then referenced as part of the beam setup in CMT.
- Dante Attenuation Calculator (DAC) – DAC performs a laborious calculation to produce attenuator settings for the Dante diagnostic. It accesses calibration data from LoCoS and Dante setup data from the CMT experiment XML as inputs, then writes the attenuator values to a database record that can subsequently be loaded from within the CMT Dante setup UI to complete the Dante configuration.
- ARC Timing Editor (ATE) – NIF's Advanced Radiographic Capability requires special configuration of four beams and has additional integration and validation needs. Shot RIs use ATE to define and validate their ARC timing configuration, with results saved to a database table from which they can be loaded by CMT.
- Target Diagnostic Template Manager (TDTM) – TD setups can be named and saved for reuse in multiple experiments. TDTM is the manager app for those "templates".

As CMT manages the setup phase of the experiment lifecycle, Approval Manager (AppMan) drives the approval workflow, since each distinct section, or data group, of every experiment must be reviewed and approved by a subject matter expert before the experiment can be exported to the laser control system, ICCS. As a data group setup is completed in CMT, the CMT user has a UI control to submit that setup for review, which it does by sending various metadata to AppMan. AppMan manages the workflow by sending email to the identified reviewers for each data group, providing links to reports that document the configured setup, and by providing a UI through which the reviewer can approve or reject the proposed setup. The AppMan setup reports are actually provided by another background application, ShotSetupReports, which renders each report into HTML for viewing in the user's web browser or into Excel. AppMan manages an approval workflow for each data group but also an overall workflow for the experiment. Once all data groups have been approved, AppMan notifies the Shot RI so they can conduct their final overall review, and lastly the NIF Operations Manager (NOM) registers a final approval that locks the entire experiment from further change. This also enables the export control, which is selected by the NOM to export the experiment to ICCS.

The last application in the suite, ConfigChecker, supports the "Readiness" experiment lifecycle phase. Readiness is the time leading up to a shot (but following the previous shot) when the Facility is being configured per the experiment request. As requested parts are installed and configurable settings are locked in, they are all tracked electronically.

ConfigChecker is the front end for a comparison feature that at each tracked location compares the part specified in the CMT request (i.e., experiment setup) to the part registered as installed. It displays a simple green or red status indicator to show visually whether there's a match or a mismatch between requested and installed parts.

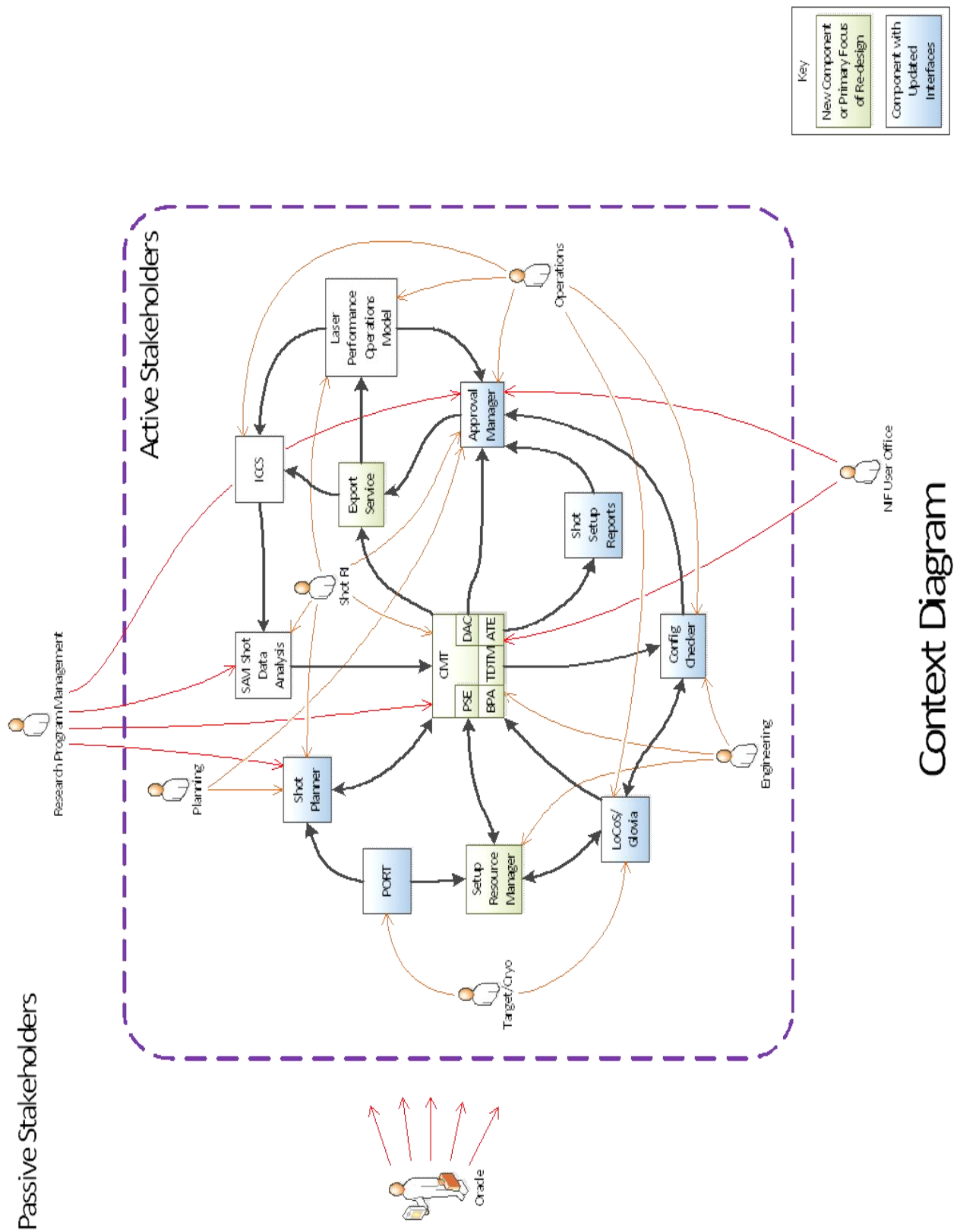


Figure 1. "To Be" System Context

System Drivers and Constraints

Operational Scenarios

1. Campaign-level operations
    a. Campaigns will no longer exist as explicit entities in the information schema. Campaigns will be, in general, sets of experiments resulting from queries over the four FLIP ID components. For each unique combination of those components, a unique Index field will distinguish the experiments in the campaign.
        i. FLIP ID Components:
            1. {PROGRAM_ABBR}
            2. {CAMPAIGN_ABBR}
            3. {PLATFORM_ABBR}
            4. {OPTIONAL_DESCRIPTOR}
    b. Creation
        i. A campaign is instantiated when the first experiment is created in it. See Experiment Creation.
            1. FLIP Campaigns (defined by participating Programs) - these represent the result of experiment queries in which each of the four FLIP ID components in the query are single-valued and program-authorized.
            2. User Campaigns – {PROGRAM_ABBR} = user OUN
            3. Virtual Campaigns – given a capability to query over internal elements of an experiment (e.g., TD participation, keywords), a virtual campaign is the non-empty result of any such query.
    c. Keywords
        i. Users may tag experiments with keywords to facilitate organization into virtual campaigns
2. Experiment-level operations
    a. Search
        i. Find existing FLIP or User Campaign experiments to load into editor
            1. Basic query interface enables pulldown selections on the four FLIP ID components plus Index. Each selection made is dynamically applied as a constraint to limit the other component selection lists such that empty campaigns are pruned from the selection space. Index selection list provides a wildcard entry.
            2. Basic text query interface supports querying by string specification composed of the three or four ID components (DESCRIPTOR optional) in the order shown in 1.a.i above, each separated by a single underscore, plus a suffix that is either an Index value or

Alias value, also separated by an underscore. The last element (Index or Alias) can be a wildcard character.

b. Load
    i. Since campaigns are realized as a particular selection across data, the editor is intrinsically experiment-oriented and able to have experiments loaded from arbitrary campaigns simultaneously
c. Create
    i. Cases
        1. Straight clone is deprecated
        2. Smart-copy: a derivative of an existing experiment with certain imposed constraints, including forced immediate resolution of stale selections, clearing some text fields, set some fields to defaults. Interface permits revisions to FLIP ID components to populate experiment into a different campaign, but defaults to the campaign of the source experiment.
        3. As-shot: a derivative of an existing experiment that has been shot; created by extracting configuration changes from the shot archive and merging them to create a new experiment. Interface permits revisions to FLIP ID components to populate experiment into a different campaign, but defaults to the campaign of the source experiment.
        4. From template:
        5. Blank experiment:
    ii. New experiment creation provides a name selection interface that offers:
        1. Selection over FLIP ID components; default selections initialized to source experiment in derivative cases
        2. Text entry (see 2.a.i.2 above) with real time validation of ID components
        3. Selection among campaigns currently loaded

3. Shot RI controls
    a. Co-RI designation – The FLIP-designated Shot RI may name others from among the pool of users holding Shot RI authorization to be Co-RIs on an experiment. Co-RIs share all RI-level privileges on the experiment.
    b. Locking & Visibility
        i. The underlying data model and the UI will support role-based ownership at the parameter level. User's lacking the ownership role for a parameter cannot edit it.

            1. The UI will enable the user to submit a change request to users that have the ownership role.
        ii. UI will support RI edit-locking at the data group level. An edit-locked data group cannot be modified. RI edit-locking is superseded by ownership.
            1. Changes in edit-lock status are dynamically pushed to clients that have the data group loaded in an editor.
        iii. Shot RI has a Participation Lock on the experiment that when set precludes the addition of new data groups. Setting it does not preclude edits to existing participating data groups.
        iv. Shot RI may set a Visibility flag on the experiment to hide an experiment from general view
        v. Archive flag removes an experiment from visibility except any view of archived experiments
4. Change serialization
    a. Replace current CMT checkout/edit/checkin model with optimistic locking. Saves are performed at data group granularity using dirty flags for each data group.
5. Data Group Metadata
    a. Data group version
6. Integration with external data sources to facilitate setup
    a. Views
        i. Shot Planner data for same experiment
        ii. Proposal Tool data for same experiment
        iii. Related documents accessed via Windchill
        iv. Target metrics; initially design values then calibrated values when available
        v. TD documents including physical envelope and output descriptions
        vi. Pulse shapes
    b. Interactions
        i. Related data source differencing – comparisons to external data sources for which a mapping to CMT experiment data is defined (Shot Planner, Proposal Tool)
        ii. Pulse shape differencing
        iii. Import fields owned & maintained in external sources (Shot Planner, Proposal Tool) that directly map into experiment setup parameters
        iv. Export fields owned in CMT that map to representations in external data sources

7. Setup selection data
    a. In a new application to succeed Parts and Lists Manager, the Setup Resource Manager (SRM), create tailored views of per-diagnostic setup data lists to enable TD Responsible System Engineers (RSE) to manage their diagnostics' options.
        i. Provide activation date option to enable automated activation at a future date
        ii. Report stale data created by changes
    b. Submit new selection option requests from within CMT
        i. For any selection, enable user to request a new part be manufactured and enabled in the setup
            1. Specification may be via free text entered directly or by a file upload, either of which may be revised arbitrarily until finally submitted.
            2. When user is satisfied with request, a control enables user to submit the request. It is then registered in Setup Resource Manager associated with the applicable selection list; notification is made to the RSE.
            3. RSE evaluates request and communicates with requestor as needed
                a. If determination is made that an existing option satisfies request, RSE resolves request by designating the chosen option. This includes outstanding part build requests still to be completed and entered into inventory.
                b. If determination is made that a new part is to be manufactured, application links enable RSE to initiate the build request. Fulfillment of the build request will be electronically communicated back to SRM, with the completed new part designated as the option satisfying the request
                c. If determination is made that no action will be taken, RSE may close request
            4. Request fulfillment in SRM is automatically propagated back to CMT, which updates the experiment without user intervention.
                a. Experiment log entry notes fulfillment
            5. A setup with an unresolved request may be submitted for review but experiments containing unresolved requests fail export.
8. Express Target Diagnostic setups

a. Each TD setup UI will be permitted to have both a low-level full-detail view (i.e., today's view) and an "express" view that maps a minimum number of input parameters into a complete setup.
    i. The express view mapping logic may populate the low-level setup directly or it may resolve to a template selection – at RS' discretion.
    ii. The express inputs should consist of primary performance characteristics of the experiment such as yield, temperature, etc – items native to the Shot RI's experiment perspective.
    iii. Express view will be the default (configurable)

9. Configurability
    a. All selections that determine the current arrangement and representation of on-screen view elements are persisted in a user-specific configuration data store
    b. In normal operation, the editor, at launch, resumes the previous session by self-configuring per the configuration data store – this includes experiment loads and navigation state, effectively managing the ongoing sequence of editing sessions as a single endless virtual editing session.
        i. This behavior is itself toggle-able as part of the configuration store. If turned off, no experiment-specific data is retained in the store.
        ii. Experiment representation does not undo changes in the experiment setup that may have been performed by other users between editing sessions. So for example, if a TD was currently in view at the end of the previous session that has since been set to de-participated, it is not re-participated.
    c. Editor supports an "override" launch mode that ignores the user's stored configuration and pushes a new configuration into use. This would enable, for example, launch of the editor in a stripped down mode for editing of a specific set of data. The new configuration is discarded at the end of the session by default but can be "popped" during the session to resume the stored configuration.

10. Persistent, comprehensive experiment history spanning activity by all users including
    a. Per-save change sets (saves occur at data group granularity); requires diff'ing against version from initial load or previous save from session; collapsible within history view.
    b. Template apply.
    c. Experiment workflow lifecycle transitions: all data group state changes and experiment-level state changes (i.e., ELMO history), export & withdraw, loads by any user.
    d. User comments (with real-time push to clients, this enables in-editor chat sessions).

    e. History interface provides content display switches such that any of the subsets of history can be removed from the current history view.
    f. All history entries are time stamped.
11. Generalized templating
    a. Super-templates can be composed from arbitrary collections of (1) other templates or (2) experiment data groups
12. Timing plot
    a. Configurable representation of any timing-relevant data from experiment: pulses, delays, detector sweeps, fiducials

Implementation Concepts and Selected and Rationale

Proposed System Operational Architecture

System Requirements

Organizational and Business Impact

Risks and Technology Readiness Assessment